Persiapan

Selanjutnya, pertama kali Anda perlu mempersiapkan segala sesuatu yang diperlukan untuk pemrograman Android, baik dari segi hardware dan software. Persiapan ini gunanya menyediakan IDE dan platform untuk pemrograman Android ini.

2.1 Hardware

Untuk melakukan pemrograman dengan Android, spesifikasi minimum hardware yang diperlukan sebagai berikut.

- Sistem operasi yang dapat Anda gunakan untuk menjalankan program Android, antara lain:
    - Windows XP (32 bit), Windows Vista (32 atau 64 bit), Windows 7 (32 atau 64 bit).
    - Mac OS X 10.5.8 atau yang lebih tinggi.
    - Linux.
- Space hard disk yang diperlukan minimal adalah 500MB (selain JDK dan Eclipse).

2.2 Software

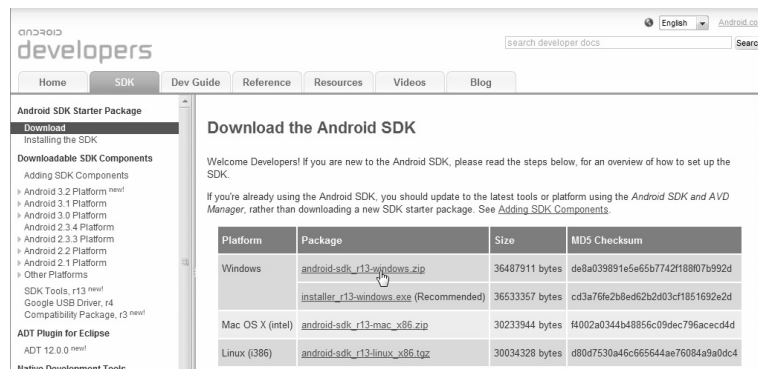
Ada tiga software utama yang perlu Anda siapkan.

- JDK (Java Development Kit)

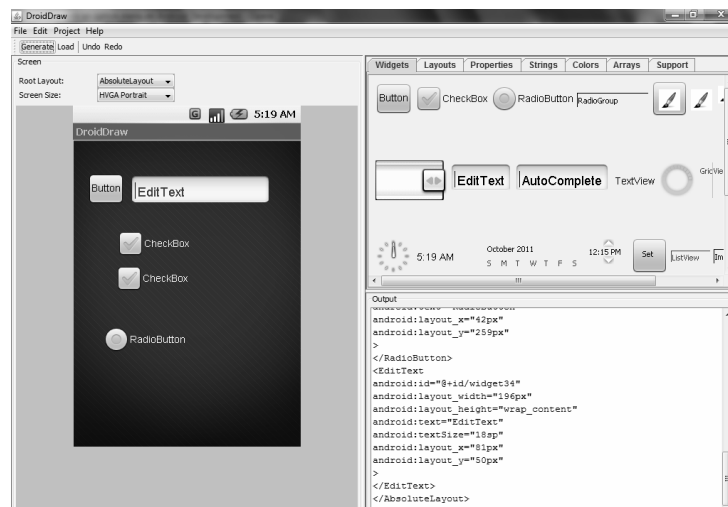
  Android ini sebenarnya berbasis bahasa Java maka kita memerlukan JDK untuk dapat men-develop aplikasi Android. JDK yang sudah termasuk di dalamnya JRE (Java Runtime Environment) untuk bisa menjalankan program yang dibuat. Anda bisa men-download JDK (sekalian termasuk JRE) dari alamat link berikut: http://www.oracle.com/technetwork/java/javase/downloads/index.html.

Gambar 2.1 Link download JDK

- SDK Android (Software Development Kit)

  Untuk men-download SDK Android, Anda bisa mengakses link http://developer.android.com/sdk/index.html. Download package terbaru pada sistem operasi yang Anda gunakan.

Gambar 2.2 Link download SDK Android

- IDE Android (Integrated Development Environment)

  Anda sebenarnya dapat menggunakan IDE untuk membangun aplikasi berbasis Android menggunakan editor teks biasa, seperti Notepad atau Wordpad atau DroidDraw.

Gambar 2.3 Tampilan DroidDraw

Namun, pada semua proyek dalam buku ini, kita akan menggunakan IDE Eclipse untuk pemrograman Android. Ada beberapa alasan yang menjadi pertimbangan, kenapa kita menggunakan IDE Eclipse, antara lain:

- Eclipse telah menyediakan struktur dasar dalam pemrograman Android, sehingga kita tidak perlu menulis dari awal semua kode program.
- Eclipse telah mendukung Android Plugin, sehingga memudahkan kita dalam membuat, meng-compile, serta running program karena terdapat emulator Android sehingga tidak perlu mencobanya pada device yang berbasis Android. Untuk men-download IDE Eclipse, Anda bisa langsung mengakses ke alamat situsnya di: http://www.eclipse.org/downloads/

Gambar 2.4 Link download IDE Eclipse

Catatan: Semua software yang diperlukan untuk pemrograman Android dapat Anda copy langsung dari Bonus CD buku ini. Namun untuk versi terbaru, Anda bisa langsung mengakses pada alamat di atas.

2.3 Proses Instalasi

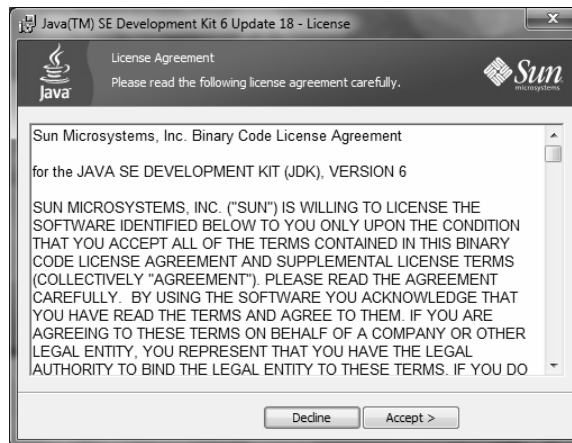
Setelah ketiga software di atas berada di komputer lokal, selanjutnya kita akan melakukan proses instalasi. Berikut ini langkah instalasi ketiga software di atas, beserta dengan settingan ketiganya agar siap digunakan untuk proses development.

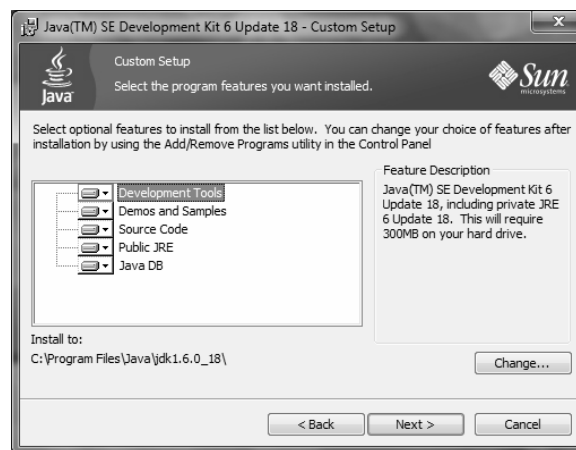
Catatan: Beberapa alamat/path yang ditampilkan dalam gambar, ada kemungkinan berbeda dengan komputer Anda, sesuaikan dengan alamat path yang ada di komputer Anda.

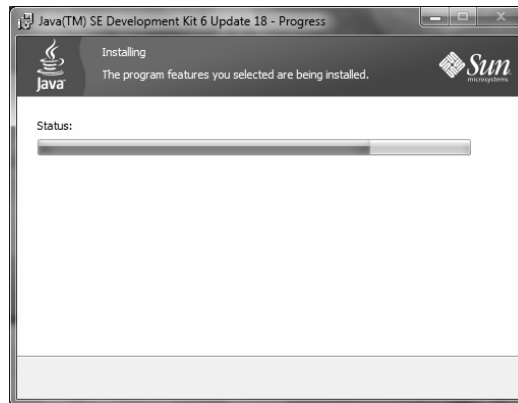
2.3.1 Instalasi JDK

Langkah instalasi JDK adalah sebagai berikut:

1. Klik dua kali pada file jdk yang Anda miliki.

2. Kemudian akan tampil kotak lisensi software, klik tombol **Accept** untuk menyetujui.

Gambar 2.5 Kotak lisensi

3. Selanjutnya Anda bisa mengatur alamat folder untuk menginstal JDK. Secara default, file tersebut akan diinstal pada folder **Program Files**. Klik tombol **Change** jika ingin mengubah alamat instalasi.

Gambar 2.6 Memilih alamat folder instalasi

4. Klik **Next**, instalasi dimulai. Tunggu sampai proses instalasi selesai.

Gambar 2.7 Proses instalasi sedang berlangsung

5. Sampai muncul pesan konfirmasi bahwa Java SDK telah berhasil diinstal.

Gambar 2.8 Pesan konfirmasi instalasi telah berhasil

6. Klik tombol **Finish** untuk mengakhiri.

Gambar 2.9 Proses instalasi selesai

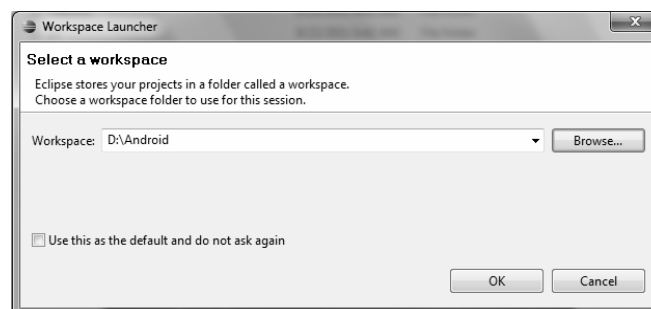
7. Anda bisa cek dan memastikan bahwa java sdk telah benar terinstal. Caranya, buka konsol command prompt dan ketikkan perintah berikut. Jika versi javac telah tampil, berarti JDK telah terinstal dengan baik.

```
C:\Program Files\Java\jdk1.6.0_18\bin>javac -version
javac 1.6.0_18

C:\Program Files\Java\jdk1.6.0_18\bin>
```

Gambar 2.10 Tes untuk memastikan JDK telah terinstal dengan baik

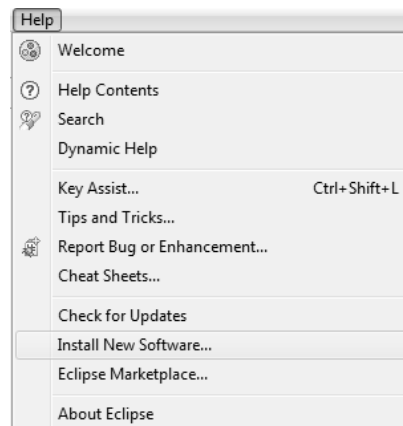
2.3.2 Instalasi Eclipse

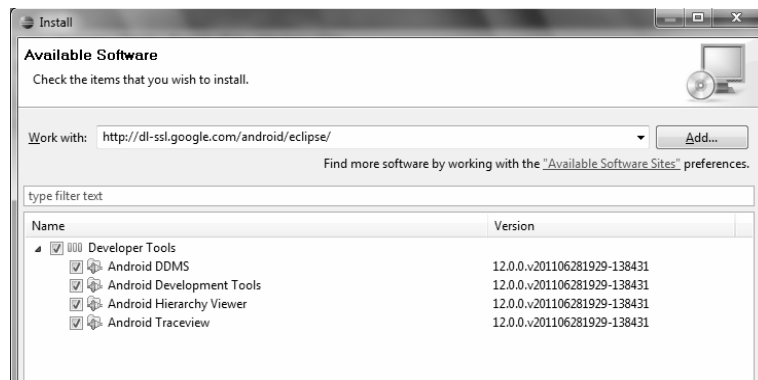
Langkah instalasi Eclipse seperti berikut ini:

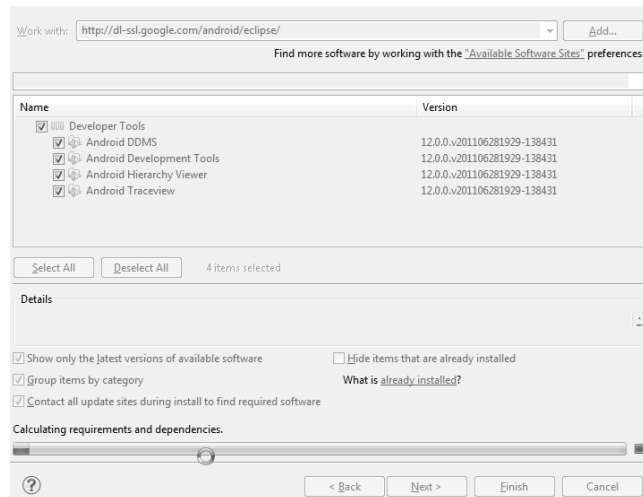
1. Proses instalasi Eclipse tergolong mudah, karena setelah Anda mendapatkan file installer Eclipse dalam bentuk terkompres, ekstrak/unkompres file tersebut ke dalam folder tertentu.
2. Selanjutnya Anda sudah mendapatkan Eclipse yang sudah siap untuk digunakan.
3. Ketika pertama kali dijalankan, Eclipse akan meminta kita menentukan di direktori mana lingkungan kerja/workspace yang digunakan. Anda bisa menentukan di mana saja sesuai keinginan.

Gambar 2.11 Menentukan alamat workspace

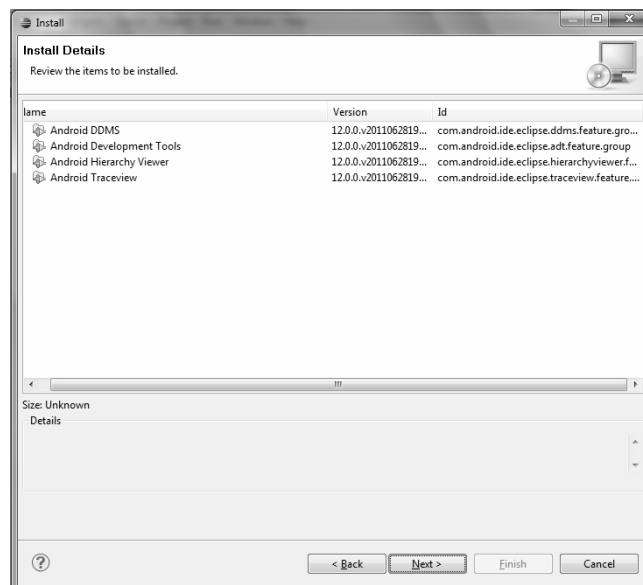
4. Untuk lebih mempermudah kita dalam proses development, kita perlu memahami interface IDE Eclipse yang

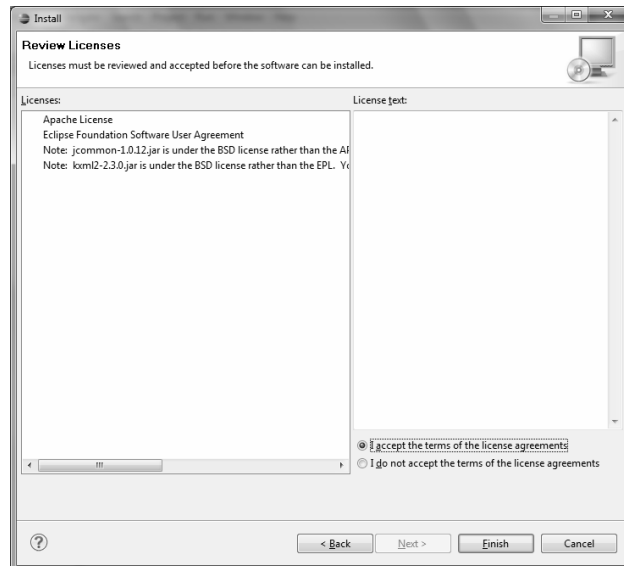
akan digunakan. Penjelasan tentang detail interface IDE Eclipse akan dijelaskan di belakang.

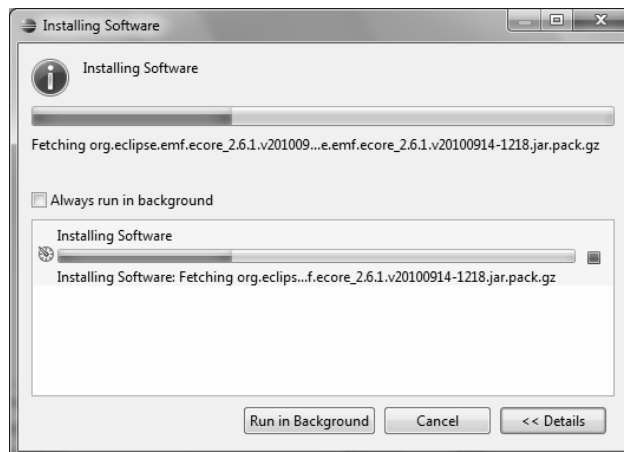
5. Untuk lebih mempermudah pemrograman Android, kita perlu menginstal *Android plugin for Eclipse*. Klik pada menu **Help** > **Install New Software**.

Gambar 2.12 Opsi Install New Software

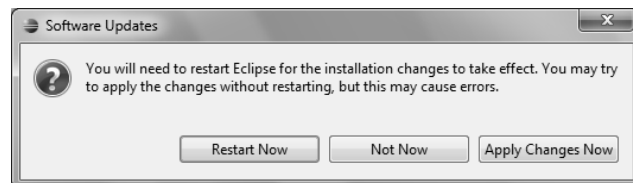
6. Akan tampil kotak **Install**, masukkan alamat berikut pada bagian **Work with**: http://dl-ssl.google.com/android/eclipse/. Klik **Add**.

Gambar 2.13 Memasukkan alamat untuk menginstal plugin Android

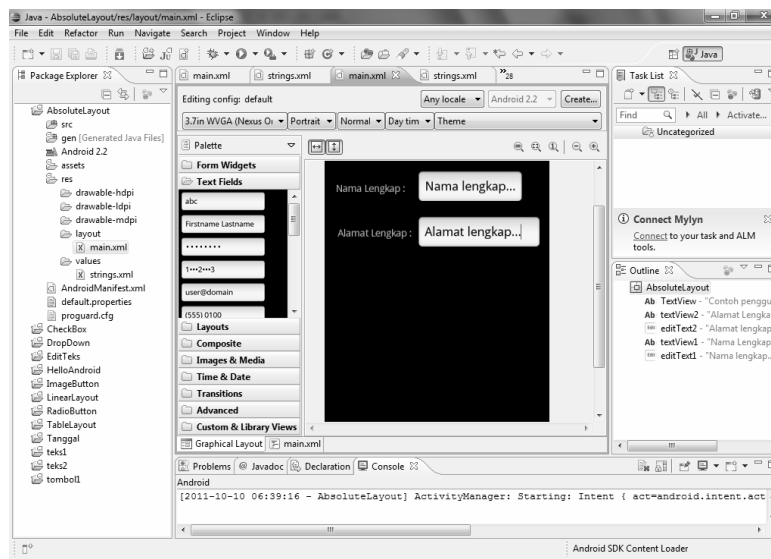
7. Aktifkan tanda centang. Klik **Next**. Tunggu sampai proses download dan instalasi selesai.

Gambar 2.14 Proses kalkulasi space dan file yang diperlukan

8. Sehingga sampai muncul pesan bahwa beberapa paket software telah siap untuk diinstal.

Gambar 2.15 Detail paket yang diinstalkan

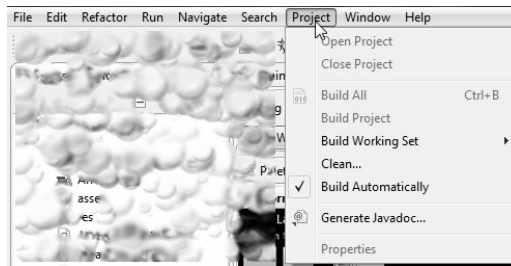
9. Klik tombol **Next** untuk melanjutkan ke proses berikutnya, kemudian klik pada bagian *I Accept the terms of the license agreement.*

Gambar 2.16 Lisensi agreement/persetujuan

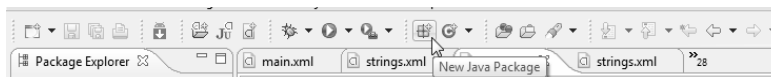
10. Klik tombol **Finish**, sehingga proses instalasi dijalankan.

Gambar 2.17 Proses instalasi software

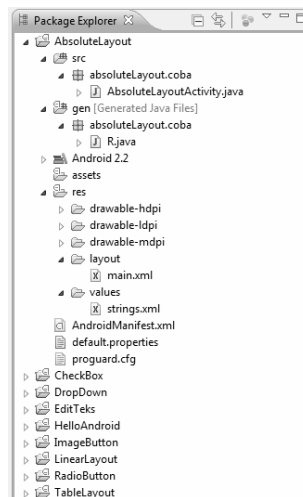
11. Setelah semua terinstal berarti *ADT Plugin* telah berhasil diinstal pada IDE Eclipse. Selanjutnya restart Eclipse Anda. Nanti juga diperlukan memberi tahu Eclipse lokasi alamat SDK Android.

Gambar 2.18 Pilihan untuk me-restart Eclipse

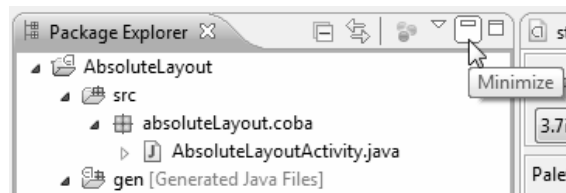
Berikut ini pengenalan IDE Eclipse yang akan Anda gunakan dalam pemrograman Android. Fitur-fitur yang ada akan memudahkan Anda melakukan pemrograman berbasis Android. Secara lengkap interface IDE Eclipse terlihat seperti berikut.

Gambar 2.19 Interface Eclipse

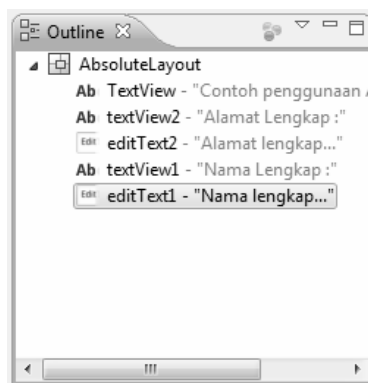
Berikut ini penjelasan bagian-bagian interface IDE Eclipse:

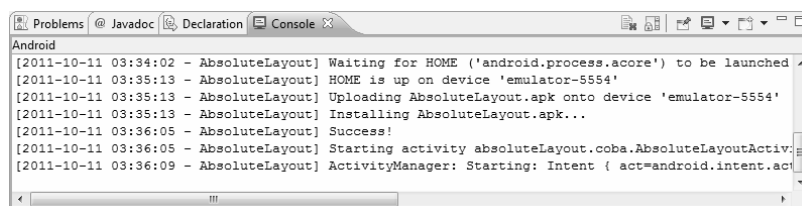
- Menu: berisi menu serta submenu yang ada pada IDE Eclipse, digunakan untuk menjalankan eksekusi perintah tertentu.

Gambar 2.20 Daftar menu pada Eclipse

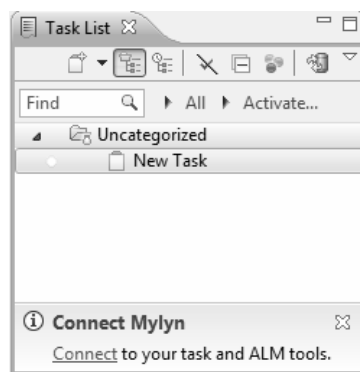
- Toolbar: berisi ikon-ikon perintah yang akan mempercepat melakukan eksekusi perintah pada Eclipse, jika dibandingkan Anda melakukan klik pada menu. Dekatkan pointer mouse Anda, maka akan terlihat tooltip yang akan tampil, menunjukkan perintah yang bisa dilakukan.

Gambar 2.21 Tooltip pada ikon di toolbar

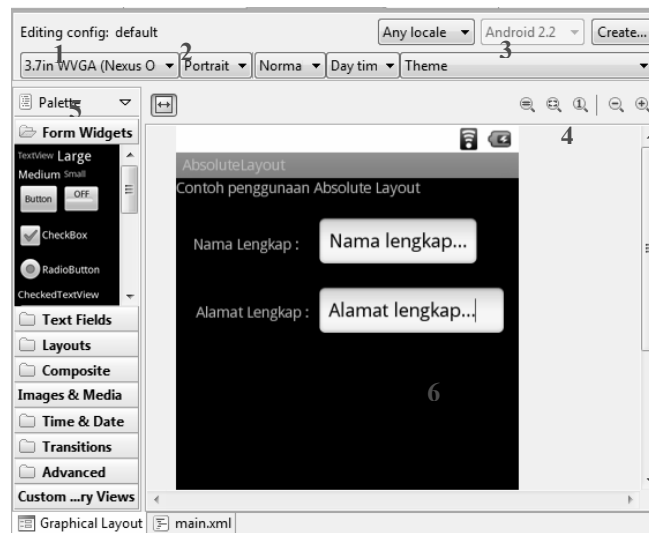
- Package Explorer: berisi paket-paket aplikasi yang dibuat. Penjelasan sub folder dan apa saja yang ada di dalamnya akan dipaparkan lebih lanjut pada bab selanjutnya.

Gambar 2.22 Jendela Package Explorer

Anda bisa meminimalkan jendela Package Explorer agar tampilan editor di IDE Eclipse lebih lebar. Klik ikon minimize yang ada di sebelah kanan atas dari jendela Package Explorer.

Gambar 2.23 Ikon minimize jendela package Explorer

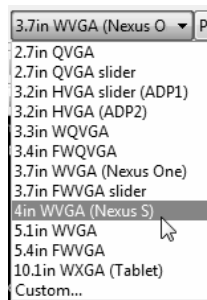
- Outline: berisi daftar widget yang digunakan dalam aplikasi.

Gambar 2.24 Jendela outline

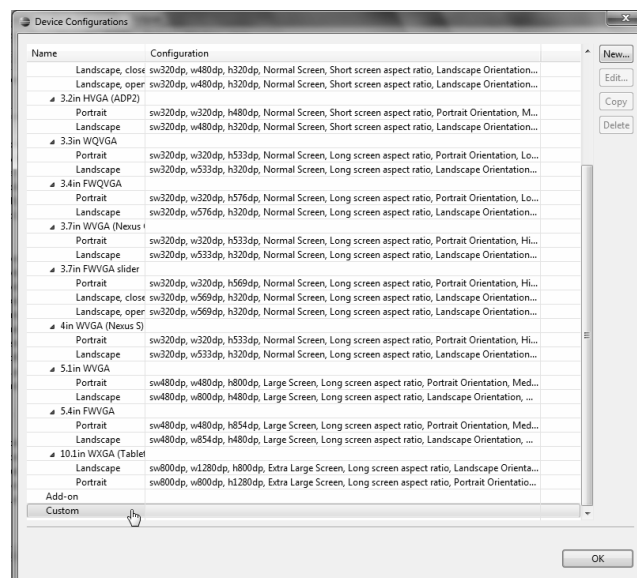
- Console: berisi informasi status program yang dijalankan. Jika terdapat error maka pesan kesalahan akan ditampilkan pada kotak ini.

Gambar 2.25 Jendela Console

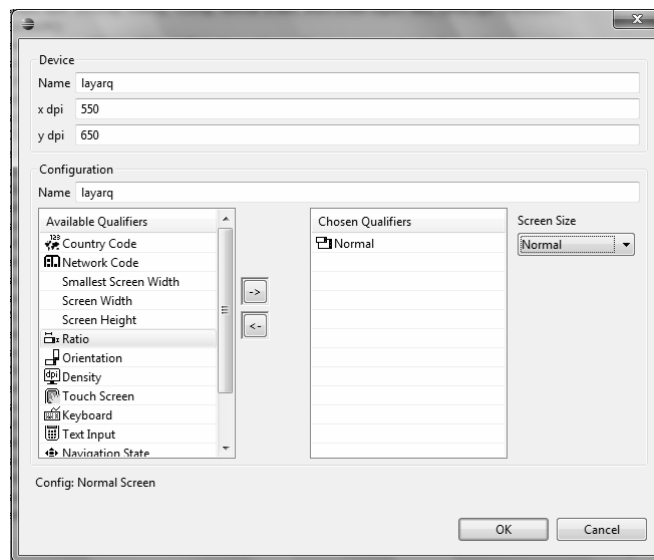
- Task List: berisi daftar task/jadwal task yang akan dilakukan, misalnya seperti instal SDK. Selain langsung koneksi ke server Google, Anda juga bisa menggunakan repository yang telah Anda siapkan sebelumnya (termasuk file-file yang dibutuhkan).

Gambar 2.26 Jendela Task List

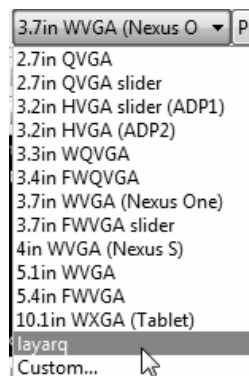
- Graphical Layout: editor grafik di mana Anda bisa memasukkan widget serta mengaturnya secara visual.

Gambar 2.27 Jendela Graphical Layout

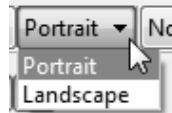
Ada beberapa bagian di jendela Graphical Layout yang bisa Anda gunakan dalam proses coding di Android. Lihat di gambar di atas.

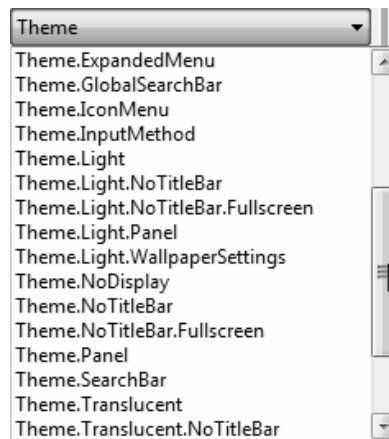
- Nomor 1: digunakan untuk konfigurasi jenis layar yang bisa Anda gunakan sebagai simulasi ketika melakukan pemrograman.

Gambar 2.28 Jenis layar yang tersedia

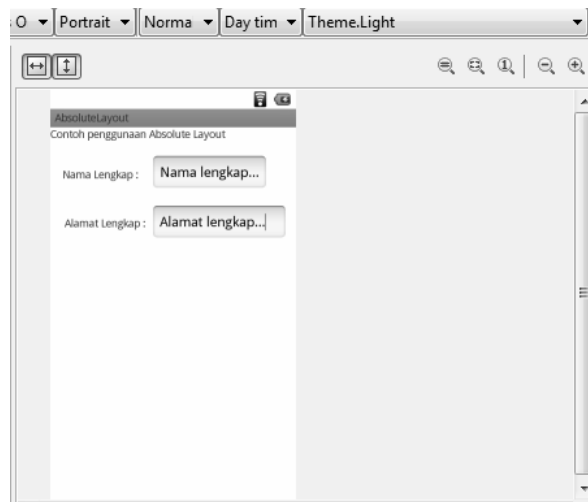
Anda bisa mengatur, bahkan membuat sendiri konfigurasi layar yang akan digunakan. Klik pada menu **Custom** sehingga tampil kotak dialog **Device Configurations**, pilih **Custom** kemudian klik tombol **New**.

Gambar 2.29 Daftar konfigurasi layar

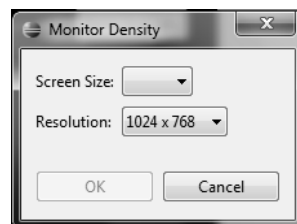
Masukkan nama berikut dengan lebar/panjang layar yang akan digunakan. Jika sudah selesai, klik tombol **OK** untuk mengakhiri proses pembuatan.

Gambar 2.30 Membuat konfigurasi layar sendiri

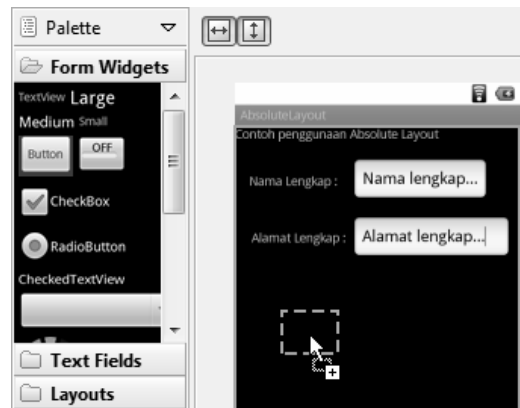
Pada daftar jenis layar tersebut akan ditambahkan nama layar baru yang sudah kita buat tadi, Anda bisa menggunakannya dalam proses coding.

Gambar 2.31 Jenis layar baru sudah ditambahkan

- Nomor 2: digunakan untuk mengatur posisi layar yang digunakan, apakah vertikal atau horizontal. Ada dua pilihan yang bisa Anda pilih, yaitu *Portrait* dan *Landscape*.

Gambar 2.32 Pilihan mengatur posisi layar

- Nomor 3: digunakan untuk mengatur theme yang dipakai pada layar visual editor. Ada banyak pilihan yang bisa Anda pilih.

Gambar 2.33 Pilihan theme

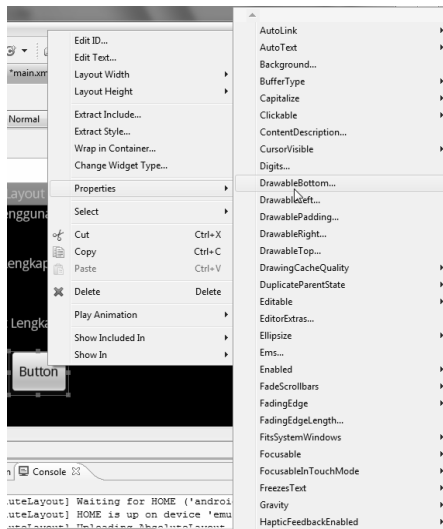
Pilih salah satu theme dan lihat perubahan apa yang terjadi pada layar visual. Misal, kita pilih **Theme.Light** maka warna layar background akan putih, terlihat cerah seperti ada cahaya.

Gambar 2.34 Penggunaan theme Light

- Nomor 4: digunakan untuk memperbesar atau memper-kecil layar visual editor. Ada beberapa pilihan yang bisa digunakan, seperti:
    - Zoom In: untuk memperbesar layar visual.
    - Zoom Out: untuk memperkecil layar visual.
    - Reset Zoom to 100% (1): memperbesar layar visual sampai 100 %.
    - Zoom to Fit (0): memperkecil layar visual sampai 0% atau kembali ke besar semula.
    - Emulate Real Size: mengatur layar visual sesuai dengan ukuran gadget yang akan digunakan. Atur berapa inci serta resolusi layar yang digunakan.

Gambar 2.35 Mengatur densitas monitor

- Nomor 5: merupakan jendela Palette yang digunakan untuk menampilkan daftar widget yang akan digunakan. Anda tinggal *drag-and-drop* widget yang digunakan ke dalam editor visual.

Gambar 2.36 Drag and drop widget dari palette ke jendela visual

- Nomor 6: merupakan jendela visual, di mana Anda bisa meletakkan widget yang digunakan dalam aplikasi. Dengan penggunaan visual editor ini, Anda bisa melihat gambaran tampilan di layar monitor Android nantinya.

Gambar 2.37 Jendela visual editor

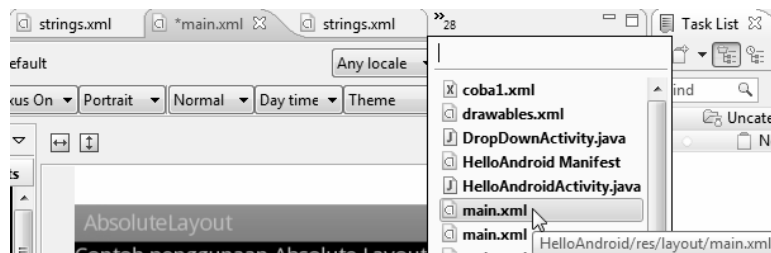
Berbagai properties yang ada pada setiap widget dapat Anda set dengan klik kanan pada widget bersangkutan dan pilih **Properties**. Salah satu kelebihannya, Anda tidak perlu menuliskan kode program, hanya perlu klik mouse saja.

Gambar 2.38 Pilihan properties pada widget

File XML: merupakan editor teks yang kode pemrograman di dalamnya di-generate secara otomatis dari visual editor. Selain melakukan modifikasi di visual editor, Anda juga bisa langsung mengubahnya di jendela editor teks.

```
<?xml version="1.0" encoding="utf-8"?>
<AbsoluteLayout xmlns:android="http://schemas.android.com/apk/res/a
    android:layout_width="match_parent" android:layout_height="matc
<TextView
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:text="@string/hello"
    />
<TextView
    android:id="@+id/textView2"
    android:text="Alamat Lengkap :"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_x="19dp"
    android:layout_y="116dp">
</TextView>
<EditText android:id="@+id/editText2" android:text="Alamat lengkap.
    <requestFocus></requestFocus>
</EditText>
<TextView android:id="@+id/textView1" android:text="Nama Lengkap :"
<EditText android:id="@+id/editText1" android:text="Nama lengkap...
<Button android:layout_height="wrap_content" android:text="Button"

</AbsoluteLayout>
```

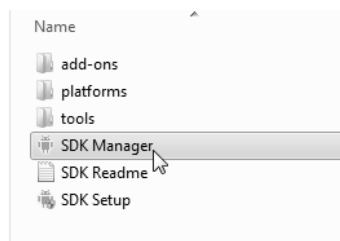
Graphical Layout | main.xml

Gambar 2.39 Editor teks

List file XML: merupakan bar yang ada di atas visual editor yang menunjukkan daftar file xml yang sedang aktif. Pada bagian sebelah kanan terdapat ikon >> yang jika didekati dengan pointer mouse akan muncul tooltip ShowList, berisi berbagai file xml yang ada. Anda bisa menggunakan daftar tersebut untuk membua file xml yang diinginkan.

Gambar 2.40 Daftar file XML

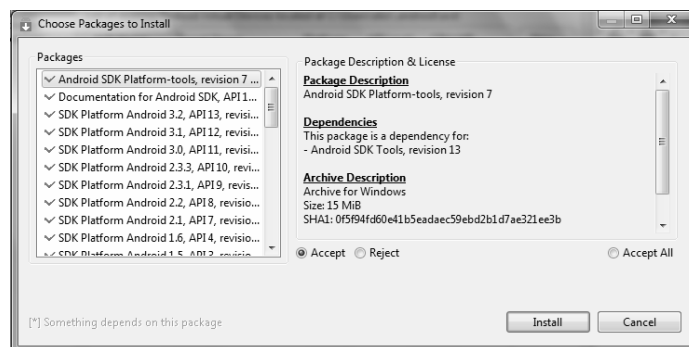
2.3.3 Instalasi SDK Android

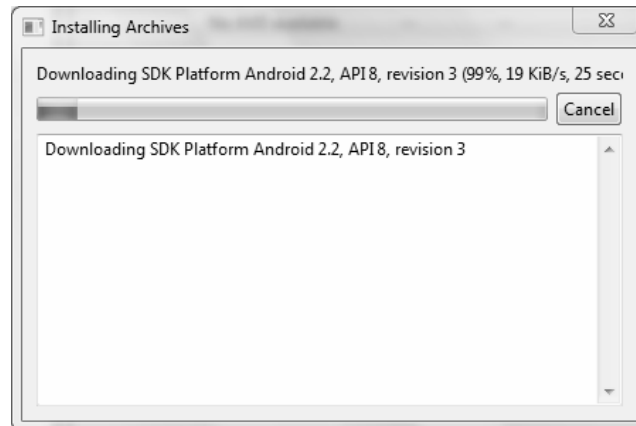
SDK android ini perlu diinstal dengan cara seperti berikut:

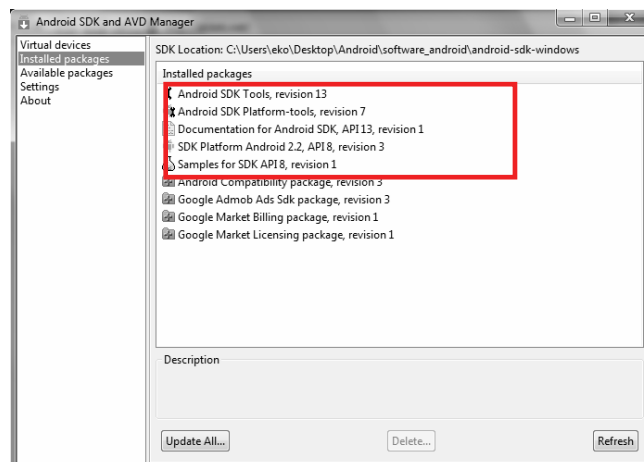
1. Setelah Anda men-dowload SDK Android dari alamat situs di atas, ekstrak ke folder tertentu dalam komputer Anda dan selanjutnya klik dua kali pada file SDK Manager.

Gambar 2.41 Isi folder sdk windows

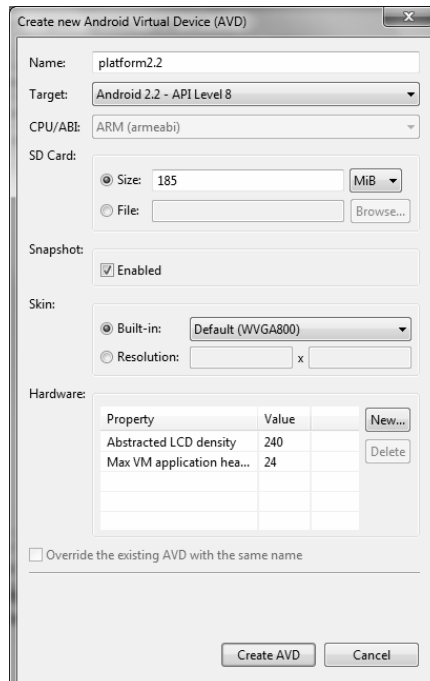
2. Klik dua kali pada file SDK Manager, tunggu sampai SDK Manager memeriksa paket apa saja yang dibutuhkan.

Gambar 2.42 SDK Manager memeriksa paket yang dibutuhkan

3. Kemudian akan muncul kotak dialog paket-paket yang akan diinstal. Pilih dan setujui satu platform saja seperti yang telah direkomendasikan oleh SDK Manager.

Gambar 2.43 Pilihan paket yang akan diinstal

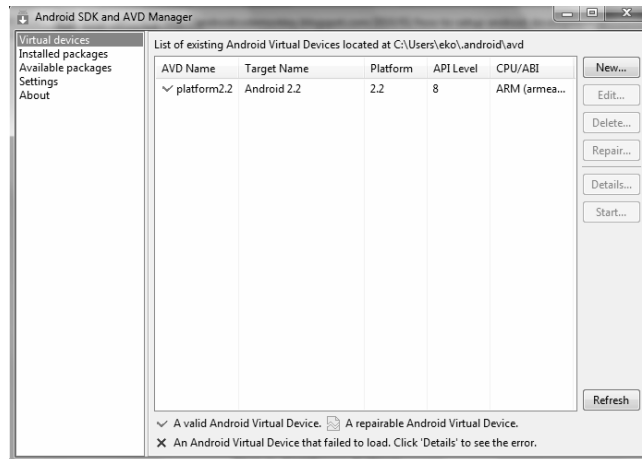
4. Setelah Anda memilih salah satu platform yang akan digunakan, klik tombol **Install** untuk memulai proses download dan instal, kemudian tunggu sampai proses selesai.

Gambar 2.44 Proses download dan install Android SDK Platform

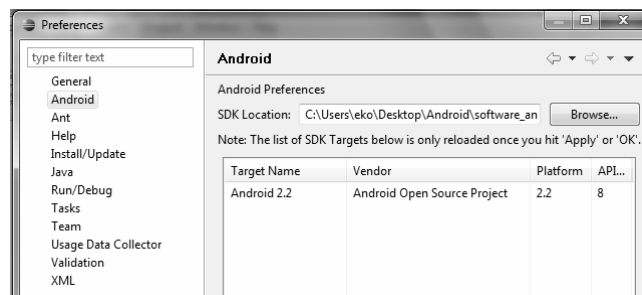
5. Setelah selesai, pada bagian **Installed packages** sudah ada paket-paket yang diinstal. Sebenarnya, minimal ada tiga paket saja yang harus ada, seperti gambar berikut ini. Dokumentasi hanya bersifat opsional, dan bisa saja tidak Anda instal.

Gambar 2.45 Paket yang diinstal

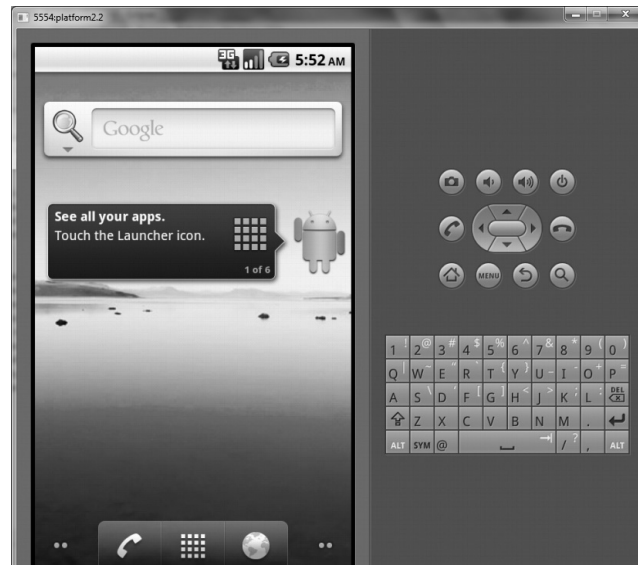
6. Selanjutnya, kita bisa membuat emulator sebuah virtual device. Klik pada menu **Virtual Device** > **New**.

Gambar 2.46 Membuat AVD baru

7. Klik tombol **Create AVD** sampai tampil keterangan bahwa AVD telah berhasil dibuat.

Gambar 2.47 Konfirmasi AVD telah berhasil dibuat

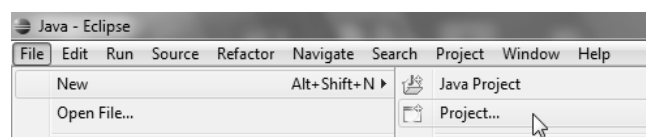
8. Sehingga pada kotak dialog **Android SDK and AVD Manager** telah muncul AVD yang barusan kita buat berdasarkan platform Android yang kita pilih.

Gambar 2.48 Tampilan AVD di kotak dialog Android SDK

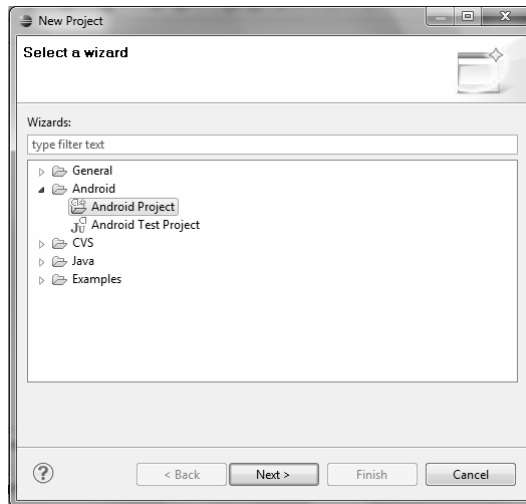
9. Selanjutnya tutup kotak setup SDK dan AVD Manager lalu buka IDE Eclipse. Beri tahu di mana lokasi SDK Android. Buka pada menu **Preferences**. Browse di mana Anda meletakkan lokasi SDK Android.

Gambar 2.49 Memberi tahu Eclipse di mana lokasi SDK Android

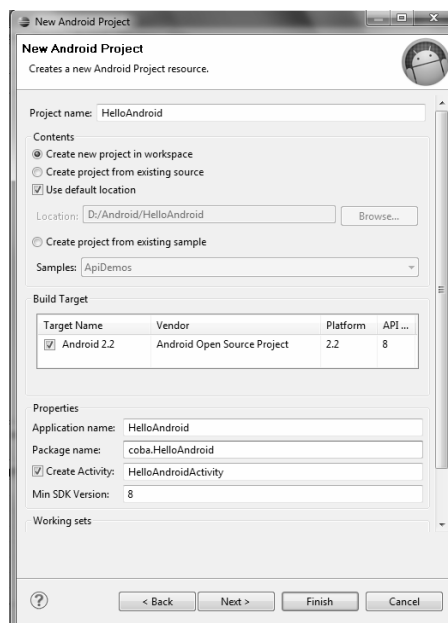
10. Maka virtual device android terbuat seperti berikut ini.

Gambar 2.50 AVD yang telah berhasil dibuat

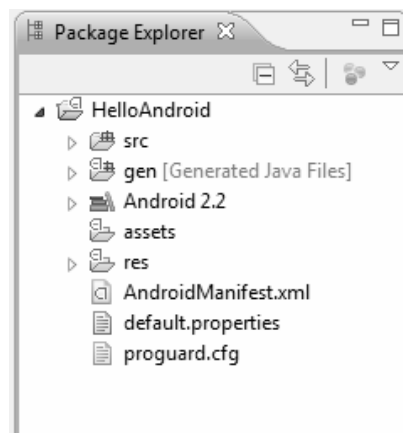
2.4 Program Hello World

Jika settingan ketiga software di atas telah benar, Anda bisa membuat aplikasi pertama, yaitu aplikasi Hello World. Di mana kebanyakan para programmer pemula memulai belajar pemrograman dengan contoh aplikasi ini. Langkah-langkahnya sebagai berikut:

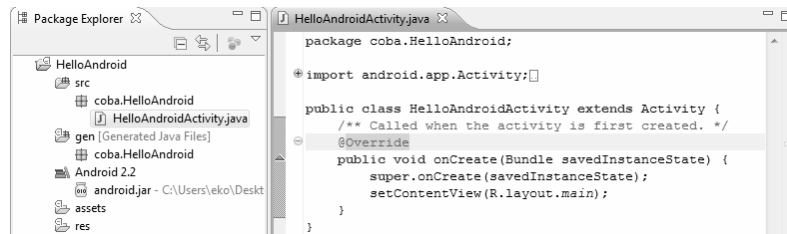
1. Klik pada menu **File** > **New** > **Project**.

Gambar 2.51 Menu File > New > Project

2. Akan tampil kotak dialog **New Project**. Pilih pada bagian **Android Project**.

Gambar 2.52 Memilih Android Project

3. Klik tombol **Next** untuk ke proses berikutnya sehingga tampil kotak dialog **New Android Project**.

Gambar 2.53 Kotak dialog New Android Project

4. Masukkan pada bagian:
    - **Project name**, nama project Android yang akan dibuat.
    - Pada bagian **Contents**, centang pada bagian **Create new project in workspace** dan centang **use default location**.
    - Pada **Build target**, centang pada AVD atau target versi Android, di mana aplikasi Anda nanti akan berjalan.
    - Kemudian isi pada bagian **Application name**, **Package name** dan centang **Create Activity**.
5. Contohnya seperti terlihat di atas, kemudian klik **Finish** jika sudah selesai.
6. Setelah Anda klik tombol **Finish** maka Eclipse secara otomatis akan membentuk sebuah folder baru berisi berbagai file pendukung yang digunakan untuk membuat aplikasi baru.

Gambar 2.54 Tree project aplikasi

Untuk bab ini, kita akan langsung membuat aplikasi HelloWorld, sedangkan penjelasan berbagai folder dan subfolder yang ada di dalamnya, akan dipaparkan pada bab selanjutnya, tentang struktur proyek Android di Eclipse.

1. Buka file HelloAndroidActivity.java yang ada di bawah src/coba.HelloAndroid.

Gambar 2.55 Isi file HelloAndroidActivity.java

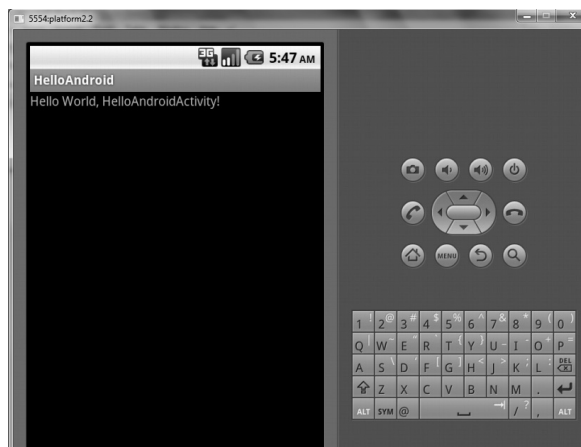
2. Selanjutnya, Anda bisa membuka file main.xml yang ada pada folder /res/layout/ atau file AndroidManifest.xml.

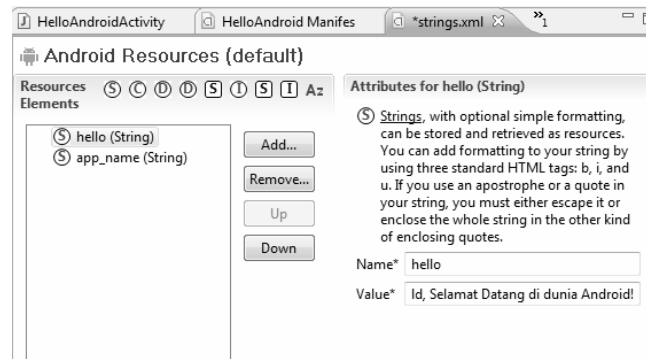
```
HelloAndroidActivity.java   HelloAndroid Manifest   main.xml
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
      package="coba.HelloAndroid"
      android:versionCode="1"
      android:versionName="1.0">
    <uses-sdk android:minSdkVersion="8" />

    <application android:icon="@drawable/icon" android:label="@strir
        <activity android:name=".HelloAndroidActivity"
                  android:label="@string/app_name">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />
                <category android:name="android.intent.category.LAUN
            </intent-filter>
        </activity>

    </application>
</manifest>
```

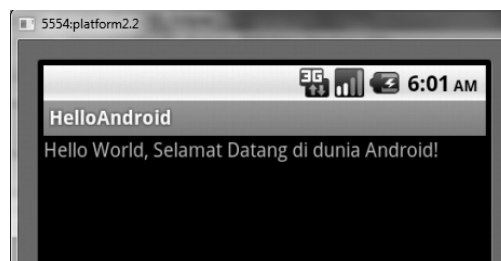
Gambar 2.56 Isi file AndroidManifest.xml

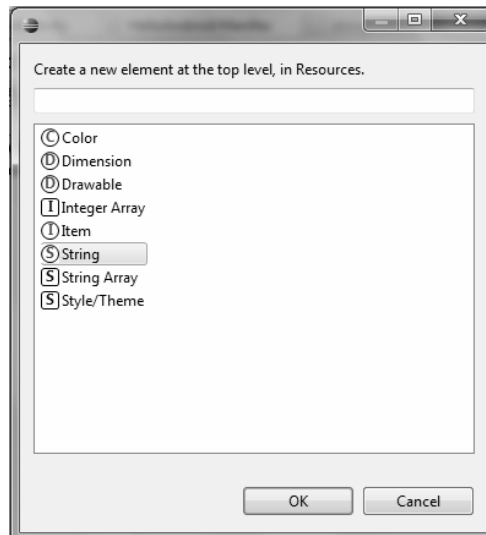
3. Untuk menjalankan aplikasi pada AVD, ada tiga cara yang bisa Anda lakukan, yaitu:
   - Pilih menu **Run** > **Run**.
   - Menggunakan tombol kombinasi **Ctrl** + **F11**.
   - Menggunakan ikon **Run** yang ada pada toolbar.
4. Jalankan salah satu perintah di atas, sehingga akan tampil kotak dialog **Run As**, dan pilih pada **Android Application**.

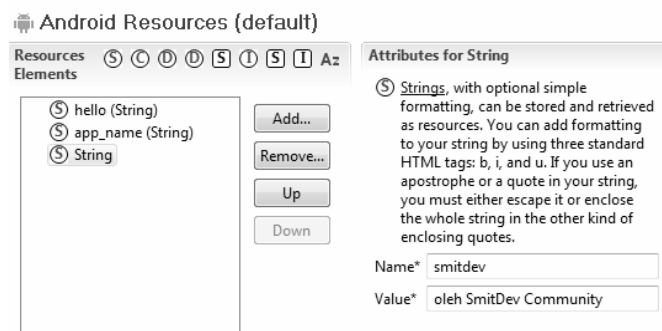
Gambar 2.57 Kotak dialog Run As

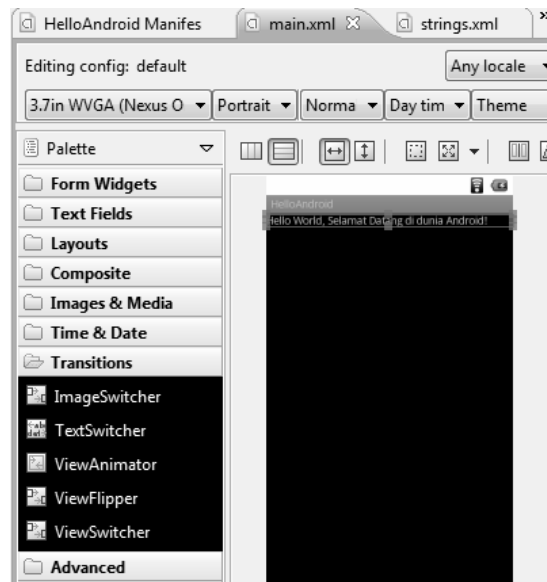
5. Tunggu proses load AVD selesai sehingga aplikasi dapat dijalankan seperti gambar berikut ini.

Gambar 2.58 Aplikasi HelloWorld sudah berhasil dijalankan

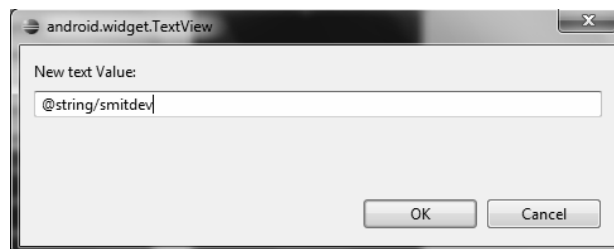
6. Kemudian coba buka file strings.xml pada folder res/value/ strings.xml. Anda bisa mengganti nilai **Value** yang ada pada string **hello**. Tekan kombinasi tombol **Ctrl+S** untuk menyimpan perubahan.

Gambar 2.59 Mengganti nilai string hello

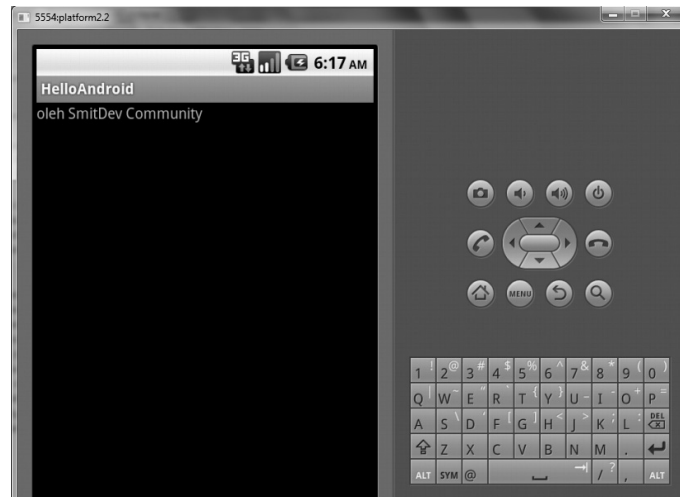
7. Jalankan aplikasi di atas sehingga akan tampil seperti berikut ini.

Gambar 2.60 Mengubah nilai string pada aplikasi

8. Anda juga bisa menambahkan string baru pada contoh aplikasi di atas. Buka kembali file strings.xml yang ada pada folder res/value/strings.xml.
9. Kemudian klik tombol **Add** untuk menambahkan string baru. Pada kotak dialog, pilih **String**, kemudian tekan tombol **OK**.

Gambar 2.61 Menambahkan elemen baru

10. Masukkan nilai baru pada bagian **name** dan **value**, yang Anda inginkan. Tekan kombinasi tombol **Ctrl+S** untuk menyimpan perubahan.

Gambar 2.62 Memasukkan value baru

11. Selanjutnya buka file res/layout/main.xml. Pada layout, klik kanan pada tulisan "Hello World, Selamat Datang di dunia Android" dan pilih **Properties** > **Text**.

Gambar 2.63 Graphical layout main.xml

12. Ubah nilai @string/hello menjadi @string/smitdev. Klik tombol **OK** untuk menerapkan perubahan.

Gambar 2.64 Mengubah nilai textview

13. Simpan file dan jalankan aplikasi tersebut maka tampilannya akan berubah menjadi seperti berikut ini.

Gambar 2.65 Menambahkan label baru pada aplikasi